

CLAIMS

What is claimed is:

1. A task management method for determining optimal placement of task components, said method comprising:

5 a) generating a communication graph representative of a task, task components represented as nodes of said communication graph and edges connecting ones of said nodes, said edges representing communication between connected nodes and being weighted proportional to communication between connected nodes;

b) assigning terminal nodes to said communication graph;

10 c) identifying nodes adjacent to terminal nodes and connected to each adjacent terminal node by a terminal edge;

d) reducing the weight of each terminal edge for each said identified node by the minimum weight of every terminal edge for said identified node;

e) determining a min cut solution for said communication graph; and

15 f) placing task components on said terminal nodes responsive to said min cut solution.

2. A task management method as in claim 1, after the step (b) of assigning terminal nodes, further comprising the step of:

20 b1) identifying independent nets in said communication graph, each of said independent nets being connected between a plurality of said terminal nodes.

3. A task management method as in claim 2, wherein the step (c) of identifying nodes comprises the steps of:

i) identifying nodes that are adjacent to each of said plurality of terminal nodes;

5 ii) selecting nodes from said identified nodes, terminal edges connected to said selected nodes having a weight greater than zero;

iii) identifying the minimum terminal edge weight for each said selected node; and

10 iv) reducing weights of all terminal edges of each selected node by its corresponding identified minimum.

4. A task management method as in claim 3, wherein each said task component is a unit of the computer program.

5. A task management method as in claim 4, wherein said each computer program unit is an instance of an object in an object oriented program

15 6. A task management method as in claim 4, wherein in step (d) computer program units are placed on computers, computer program units being placed on a common computer being combined into a single component.

20 7. A task management method as in claim 3 wherein said task is integrated circuit chip functional element placement and said task components are logic elements, said logic elements being placed on an integrated circuit chip in placement step (e).

8. A distributed processing system for determining optimal placement of computer program components on multiple computers, said distributed processing system comprising:

means for generating a communication graph of nodes interconnected by edges and representative of a computer program, computers executing said computer program being represented as terminal nodes, computer program components being represented as non-terminal nodes and edges connecting ones of said non-terminal nodes, said edges representing communication between connected nodes and being weighted proportional to communication between connected nodes;

means for identifying non-terminal nodes connected to terminal nodes;
means for identifying a minimum terminal edge weight;
means for reducing terminal edge weights responsive to said identified minimum weight;

means for determining a min cut solution for said communication graph; and
means for placing program components on ones of said computers responsive to said determined min cut solution;
said computer program being executed by said computers.

9. A distributed processing system as in claim 8, further comprising:

means for identifying independent nets connected between a plurality of said terminal nodes.

10. A distributed processing system as in claim 9 wherein said means for identifying non-terminal nodes connected to terminal nodes identifies non-terminal nodes connected to all terminal nodes connected to an independent net.

11. A distributed processing system as in claim 10, wherein each said program component is a unit of the computer program.

12. A distributed processing system as in claim 11, wherein said each program unit is an instance of an object in an object oriented program

5 13. A computer program product for determining optimal placement of functional components, said computer program product comprising a computer usable medium having computer readable program code thereon, said computer readable program code comprising:

10 computer readable program code means for generating a communication graph of nodes interconnected by edges and representative of a function, said nodes including a plurality of terminal nodes, functional components being represented as non-terminal nodes, said edges representing communication between connected nodes;

computer readable program code means for identifying non-terminal nodes connected to terminal nodes;

15 computer readable program code means for identifying a minimum terminal edge weight;

computer readable program code means for reducing terminal edge weights responsive to said identified minimum weight;

20 computer readable program code means for determining a min cut solution for said communication graph; and

computer readable program code means for placing program components on said terminal nodes responsive to said determined min cut solution.